xRSPCI

MANUEL D’UTILISATION
HARDWARE & SOFTWARE

xRSPCI

MANUEL
D'UTILISATION
HARDWARE &
SOFTWARE

MARQUES DEPOSEES ®

- *ACKSYS* est une marque déposée de *ACKSYS*.
- Windows XP, Windows 2000, Windows NT, Windows 95, Windows 98 et Windows ME sont des marques déposées de Microsoft.

NOTICE

ACKSYS ® ne garantit en aucune façon le contenu du présent document et dégage son entière responsabilité quant à la rentabilité et la conformité du matériel aux besoins de l'utilisateur.

ACKSYS ® ne pourra en aucun cas être tenu pour responsable des erreurs éventuellement contenues dans ce document, ni des dommages quelle qu'en soit l'importance, du fait de la fourniture, du fonctionnement ou de l'utilisation du matériel.

ACKSYS ® se réserve le droit de réviser périodiquement ce document, ou d'en changer le contenu, sans aucune obligation pour ACKSYS ® d'en aviser qui que ce soit.

TABLE DES MATIERES

1. INTRODUCTION

CARTE DE COMMUNICATION SERIE, 2, 4 OU 8 VOIES, PCI PLUG & PLAY.

La carte XRSPCI offre une solution fiable et performante en environnement industriel pour les applications de communication qui nécessitent :

- des liaisons de communications séries haut débit,
- des interfaces type RS232, RS422, RS485

La carte XRSPCI est conforme aux spécifications PCI 2.1 ou supérieures, ce qui rend l'installation très facile (pas de cavaliers ni d'interrupteurs pour le niveau d'interruption ou l'adresse de base). Toutes les ressources de la carte sont automatiquement attribuées par le BIOS PCI lors de la mise sous tension du P.C.

La carte est livrée avec les pilotes Windows 98, Windows ME, Windows NT, Windows 2000 & Xp et Linux (noyau 2.2 et 2.4). D'autres pilotes peuvent être développés sur demande.

Plusieurs cartes XRSPCI peuvent cohabiter au sein d'une même machine pour réaliser des configurations 8, 16, 24 ou 32 ports.

2. CARACTERISTIQUES DES CARTES DE LA GAMME XRSPCI

2.1 Caractéristiques communes aux différentes versions

- 2, 4 ou 8 ports de communication série asynchrone RS232, RS422 ou RS485
- Le jeu de registres des UARTS est compatible 16C550.
- Vitesse de communication maximale :
  - En RS232 garantie jusqu'à **250 Kbits/s**, testée jusqu'à **460 Kbits/s**
  - En RS422/485 garantie et testée jusqu'à **1,8 Mbits/s**
- Interface PCI esclave 2.1 ou supérieure.
- Registres de polling (image des huit lignes d'interruption des UARTs)
- Identification des configurations particulières de la carte par lecture d'un registre
- 4 adresses de base, BAR0 à BAR3
  - BAR0 : Adresse base des UARTs en I/O
  - BAR1 : Adresse base des UARTs en mémoire
  - BAR2 : Adresse base des registres de configuration en I/O
  - BAR3 : Adresse base des registres de configuration en mémoire

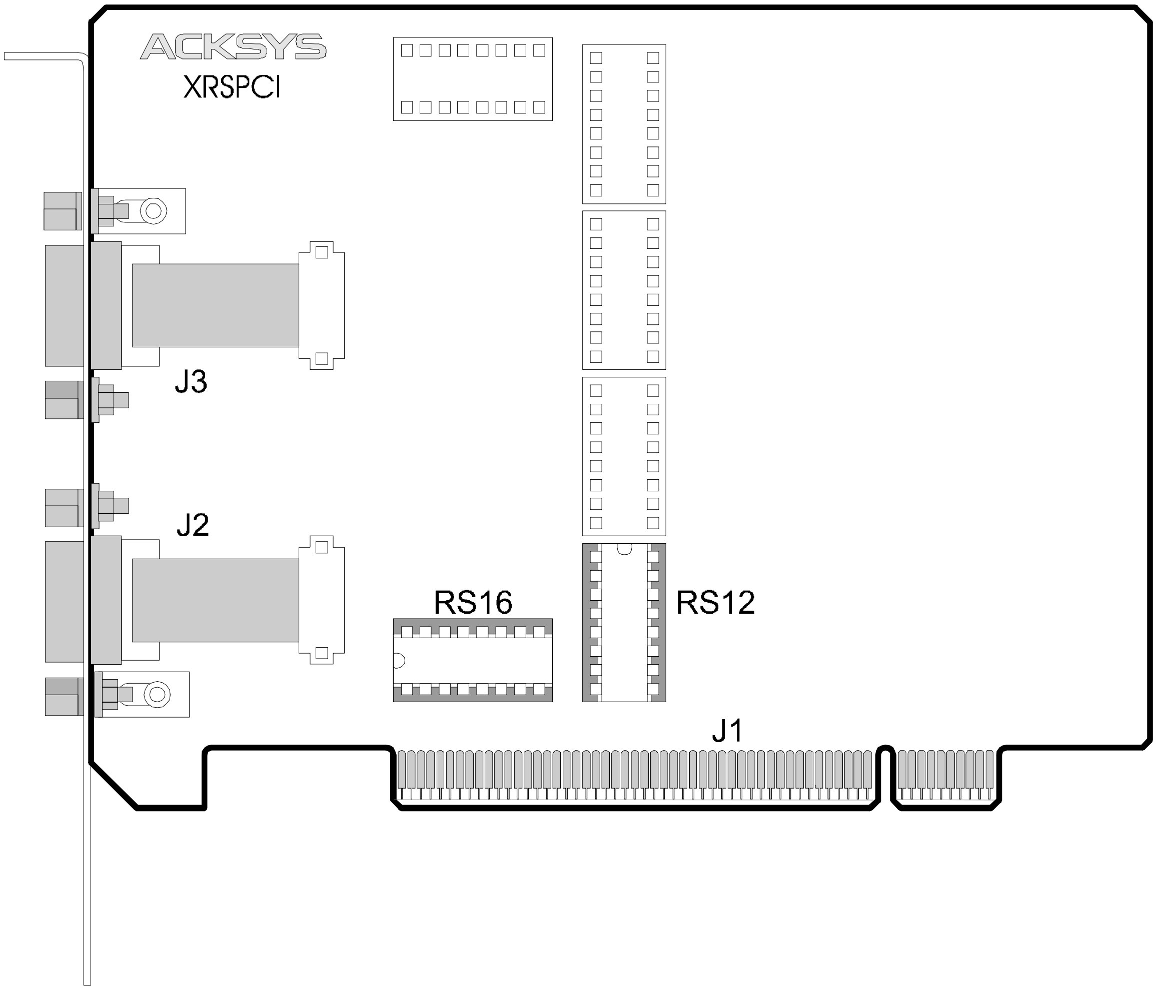
  Une description détaillée des registres de configuration est disponible sur simple demande pour les clients qui désirent développer un pilote.

2.2 Fonctions avancées de communication

- Jusqu'à 128 octets de FIFO par port, en émission et en réception.
- Interface PCI intégrée aux UARTs optimisant les transferts de données.
- Mode RS485 avec retournement automatique des drivers de ligne.
- Contrôle de flux matériel/logiciel automatique.
- Générateur de bauds avancé pour les vitesses non standards.
- Vitesse de transmission jusqu'à 1,8 Mbps (interface RS422) avec oscillateur standard 29,4912 MHz.
- Protection ESD ± 15kV sur chaque port.

2.3 Implantation connecteurs et réseaux de la carte 2RSPCI

Les réseaux ne peuvent être montés que sur les versions de cartes en RS422/485.

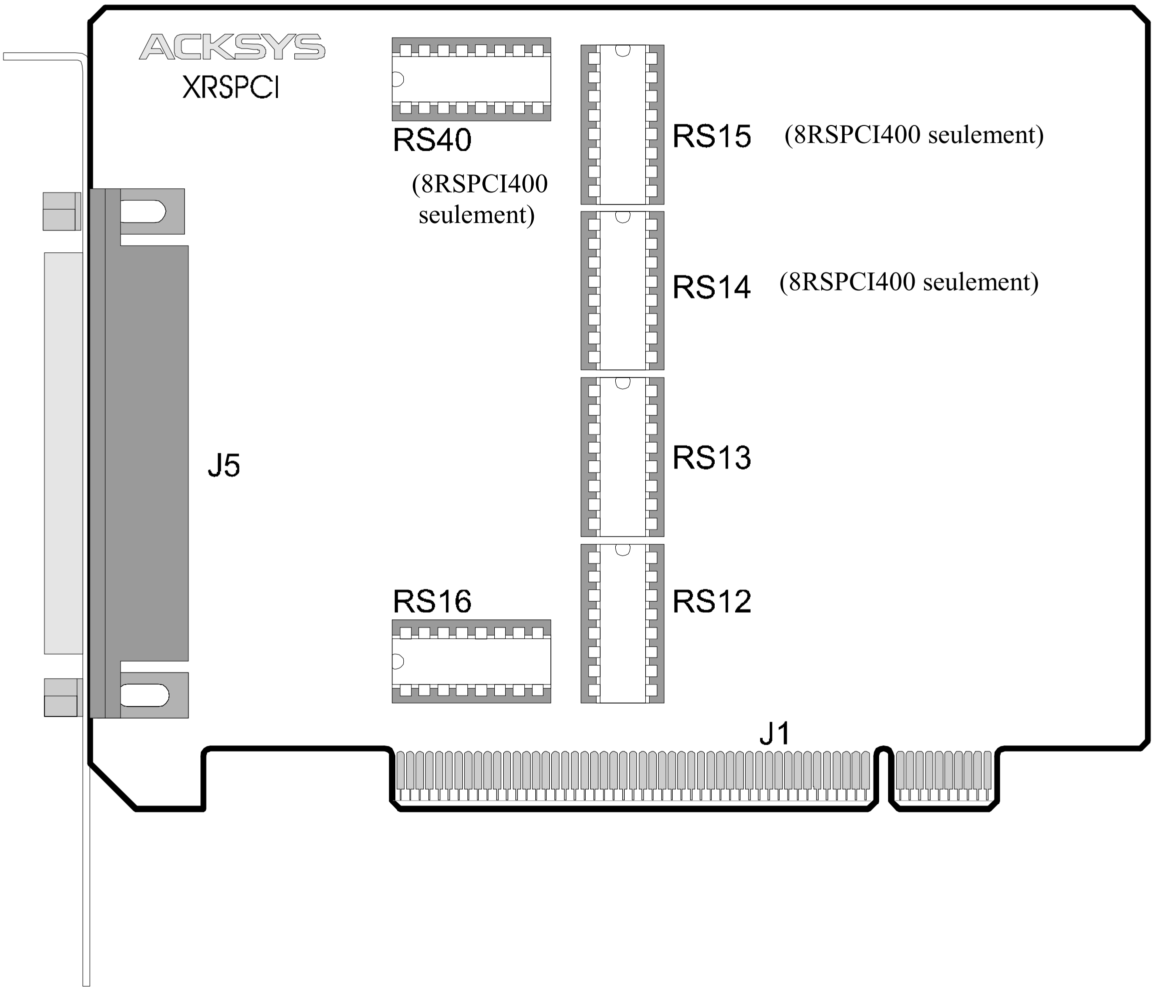


J1 : Connecteur PCI (environnement 5V uniquement)
J2 : Connecteur SUB D 9 points du port 1.
J3 : Connecteur SUB D 9 points du port 2.

Connecteurs 2RSPCI232 : male
Connecteurs 2RSPCI400 : femelle

2.4 Implantation connecteurs et réseaux des cartes 4RSPCI & 8RSPCI

Les réseaux ne peuvent être montés que sur les versions de cartes en RS422/485.

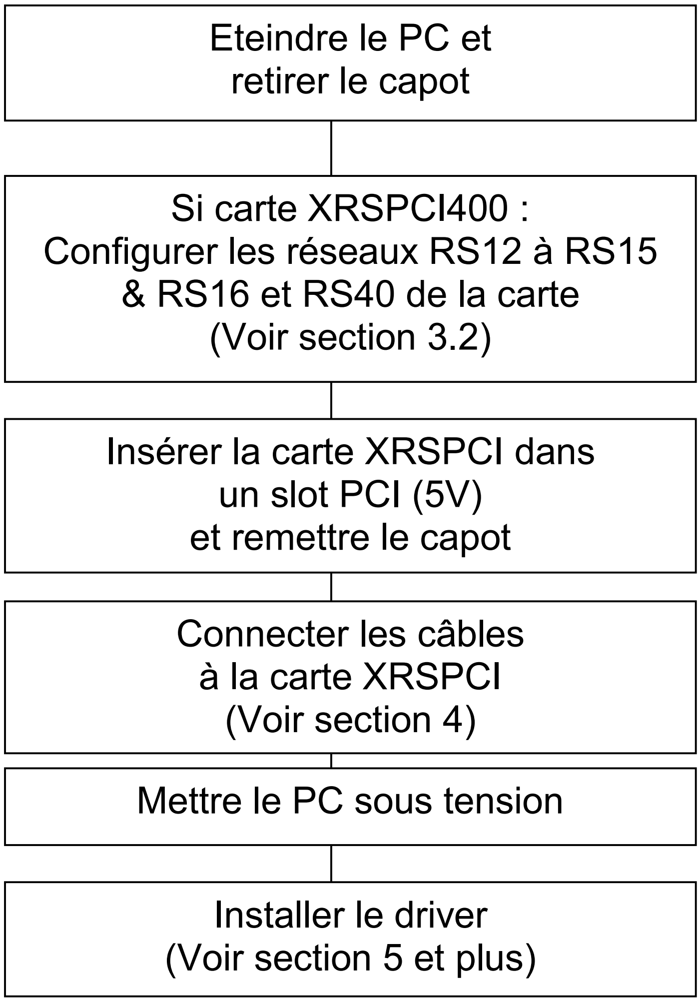


J1 : Connecteur PCI (environnement 5V uniquement)
J5 : Connecteur 62 points femelle pour câble pieuvre ACKSYS ou câble client.

Câble pieuvre ref 4P232 fourni avec la carte 4RSPCI232
4 SUB D 9 mâle
Câble pieuvre ref 4P422 fourni avec la carte 4RSPCI400
4 SUB D 25 mâle
Câble pieuvre ref 8P232 fourni avec la carte 8RSPCI232
8 SUB D 9 mâle
Câble pieuvre ref 8P422 fourni avec la carte 8RSPCI400
8 SUB D 25 mâle

3. INSTALLATION DE LA CARTE

3.1 Synoptique général



En cas de problème, se référer à la section **PROBLEMES RENCONTRES**

3.2 Configuration des réseaux RSxx

Ce paragraphe ne concerne que les cartes de la gamme xRSPCI400.
Pour ces dernières, le mode de fonctionnement en RS422 ou RS485 ainsi que le paramétrage des résistances de terminaison et polarisation sont assurés par la présence ou l'absence de réseaux montés sur support, comme indiqué dans les tableaux 3.1, 3.2 et 3.3 ci-dessous.

Mode 422/485

Les voies sont configurables en RS422 ou RS485 par bloc de 2 voies.

		2RSPCI400		4RSPCI400		8RSPCI400	
Réseau	**Ports**	**RS422**	**RS485**	**RS422**	**RS485**	**RS422**	**RS485**
RS12	**1 & 2**	○	●	○	●	○	●
RS13	**3 & 4**	-	-	○	●	○	●
RS14	**5 & 6**	-	-	-	-	○	●
RS15	**7 & 8**	-	-	-	-	○	●

Tableau 3.1 : Montage des réseaux RS12 à RS15 par type de carte

Terminaison & polarisation

Les résistances de terminaison et polarisation sont configurables par bloc de 4 voies.

		2RSPCI400 / 4RSPCI400 /8RSPCI400	
Réseau	**Ports**	**Sans polarisation & Terminaison**	**Avec polarisation & Terminaison**
RS16	**1, 2, 3 & 4**	○	●

Tableau 3.2 : Montage du réseau RS16

		8RSPCI400	
Réseau	**Ports**	**Sans polarisation & Terminaison**	**Avec polarisation & Terminaison**
RS40	**5, 6, 7 & 8**	○	●

Tableau 3.3 : Montage du réseau RS40

○ Réseau absent ● Réseau présent - Non applicable

4. CONNECTIQUE & CABLAGE

En RS232, les cartes sont toutes équipées de connecteur SUBD 9 point mâle, au nombre de 2 directement en face arrière pour les modèle 2 voies ou au nombre de 4 et 8 distribués par un câble pieuvre externe pour les modèles 4 et 8 voies.

En RS422/485, il y a 2 types de connecteurs suivant les modèles de carte.
- Les cartes 2 voies sont équipées de 2 SUBD 9 femelle directement en face arrière
- Les cartes 4 et 8 voies sont équipées de 4 et 8 SUBD 25 mâles distribués par un câble pieuvre externe.

Notez que les cartes 4 et 8 voies sont équipées d’un connecteur SUBD 62 femelle en face arrière qui peut aussi être utilisé dans le cas ou le câble pieuvre externe ne serait pas souhaité.

4.1 Mode RS232

Le tableau 4.1 donne la répartition des signaux sur les connecteurs SUBD 9 mâle en mode RS232.

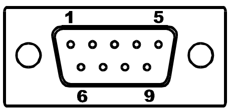
Carte 2RSPCI232 Carte 4RSPCI 232 équipée de la pieuvre 4P232 Carte 8RSPCI 232 équipée de la pieuvre 8P232			
SUB D 9 mâle	**Pin**	**Signal**	**Fonction**
 EIA/TIA 574 DTE	1	DCD (109)	Data Carrier Detect
	2	RXD (104)	Réception
	3	TXD (103)	Emission
	4	DTR (108)	Data Terminal Ready
	5	GND (102)	Masse
	6	DSR (107)	Data Set Ready
	7	RTS (105)	Request To Send
	8	CTS (106)	Clear To send
	9	RI* (125)	Ring Indicator

Tableau 4.1 : Répartition des signaux sur les connecteurs 9 points

*** ATTENTION** : LE SIGNAL RI (125) N’EXISTE QUE SUR LES CARTES 2RSPCI (2 PORTS). LA BROCHE 9 DES CONNECTEURS SUBD 9 DES PIEUVRES 4P232 & 8P232 N’EST PAS CABLEE.

4.2 Mode RS422

Le tableau 4.2 ci-dessous donne la répartition des signaux sur les connecteurs 25 points mâles de la pieuvre en mode RS422 (cartes 4 et 8 ports).
Le tableau 4.3 indique la répartition des signaux sur les connecteurs 9 points femelles des cartes 2 ports.
Les broches non représentées dans les tableaux sont inutilisées.

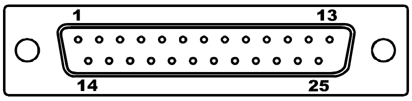
Carte 4RSPCI400 équipée de la pieuvre 4P422 en mode 422 Carte 8RSPCI400 équipée de la pieuvre 8P422 en mode 422			
SUBD 25 mâle	**Pin**	**Signal**	**Fonction**
	2	A	Emission (+Tx)
	3	A’	Réception (+Rx)
	7	GND	Masse
	9	-POL	Polarisation
	14	B	Emission (-Tx)
	16	B’	Réception (-Rx)
	19	+POL	Polarisation

Tableau 4.2 : Répartition des signaux RS422 sur les connecteurs SUB D 25

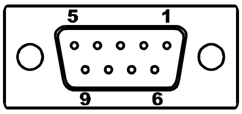
Carte 2RSPCI400 en mode 422			
SUBD 9 femelle	**Pin**	**Signal**	**Fonction**
	1	-POL	Polarisation
	2	A’	Réception (+Rx)
	3	A	Emission (+Tx)
	4	+POL	Polarisation
	5	GND	Masse
	7	B	Emission (-Tx)
	8	B’	Réception (-Rx)

Tableau 4.3 : Répartition des signaux RS422 sur les connecteurs SUB D 9

4.3 Mode RS485

Le tableau 4.4 ci-dessous donne la répartition des signaux sur les connecteurs 25 points mâles de la pieuvre en mode RS485 (cartes 4 et 8 ports).
Le tableau 4.5 indique la répartition des signaux sur les connecteurs 9 points femelles des cartes 2 ports en mode RS485.
Les broches non représentées dans les tableaux sont inutilisées.

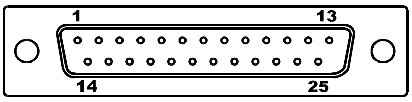
Carte 4RSPCI400 équipée de la pieuvre 4P422 en mode 485 **Carte 8RSPCI400 équipée de la pieuvre 8P422 en mode 485**			
SUBD 25 mâle	**Pin**	**Signal**	**Fonction**
	2	Réservé	
	3	AA'	Emission/Réception (+Tx/Rx)
	7	GND	Masse
	9	-POL	Polarisation
	14	Réservé	
	16	BB'	Emission/Réception (-Tx/Rx)
	19	+POL	Polarisation

Tableau 4.4 : Répartition des signaux RS485 sur les connecteurs SUB D 25

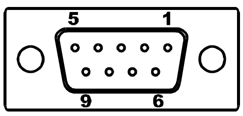
Carte 2RSPCI400 en mode 485			
SUBD 9 femelle	**Pin**	**Signal**	**Fonction**
	1	-POL	Polarisation
	2	AA'	Emission/Réception (+Tx/Rx)
	3	Réservé	
	4	+POL	Polarisation
	5	GND	Masse
	7	Réservé	
	8	BB'	Emission/Réception (-Tx/Rx)

Tableau 4.5 : Répartition des signaux RS485 sur les connecteurs SUB D 9

4.4 Recommandations de câblage en RS232

Le connecteur SUBD 9 des cartes xRSPCI en version RS232 identifie un périphérique DTE conforme à la norme EIA/TIA 574 (sauf RI pour les modèles 4 et 8 voies qui n'est pas disponible).

- Pour les connecter à un périphérique DCE (Modem), utiliser un câble droit
- Pour les connecter à un périphérique DTE, utiliser un câble croisé (null-modem)

4.5 Recommandations de câblage en RS422/RS485

Pour éviter toute inversion de câblage engendrée par l'utilisation de l'appellation d'usage + et – qui n'est pas normalisée (un constructeur appellera + ce qu'un autre appelle -), réaliser votre câble de la façon suivante :

En mode RS422 :

Identifier côté équipement les signaux A, A', B, B'
Les points A, B, A' et B' sont définies selon les recommandations EIA-422 et V11 tels que: $V_A < V_B$ et $V_{A'} < V_{B'}$ à l'état de repos, état encore appelé MARK ou OFF (Emission / Réception de bits de stop).
le signal A de la xRSPCI sur le signal A'de l'équipement
le signal B de la xRSPCI sur le signal B'de l'équipement
le signal A' de la xRSPCI sur le signal A de l'équipement
le signal B' de la xRSPCI sur le signal B de l'équipement

En mode RS485 :

Identifier côté équipement les signaux AA' et BB'
Les points AA' et BB' sont définis selon les recommandations EIA-485 et V11 tels que :$V_{AA'} < V_{BB'}$ à l'état de repos, état encore appelé MARK ou OFF (Emission / Réception de bits de stop).
le signal AA' de la xRSPCI sur le signal AA' de l'équipement
le signal BB' de la xRSPCI sur le signal BB' de l'équipement

Polarisation & résistances de terminaison :

Les résistances de polarisation et les résistances de terminaison sont intégrées sur la carte et connectées par des réseaux montés sur support.

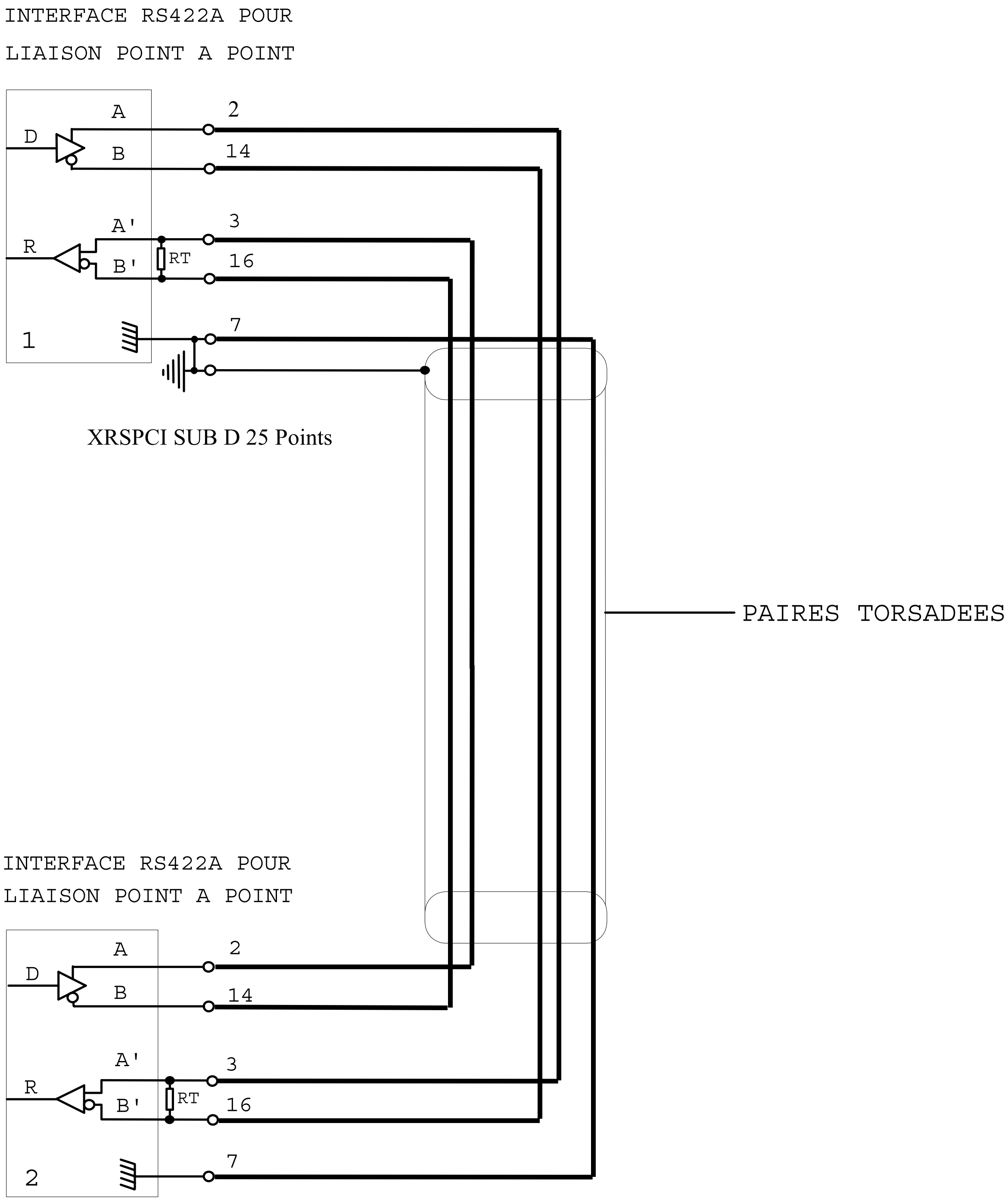
La polarisation de ligne peut être nécessaire en mode RS422 dans le cas d'un fonctionnement en maître/esclave multipoint (appelé aussi RS485 4 fils). Une seule polarisation doit être présente sur le bus.
La polarisation de ligne est toujours nécessaire en mode RS485. Une seule polarisation doit être présente sur le bus.

Les résistances de terminaison permettent de limiter les réflexions sur la ligne lorsque les distances entre équipements sont importantes.

Si la polarisation est requise uniquement sur certains ports, ou si la présence simultanée des résistances de terminaison pose un problème, il est possible de polariser les lignes individuellement par rebouclage sur le connecteur. Dans ce cas, le réseau correspondant aux ports concernés doit être ôté. En RS422, pour polariser la ligne de réception d'un port, relier sur le connecteur le signal **B'** au signal **+POL**, et relier le signal **A'** au signal **-POL**. En RS485, pour polariser la ligne de réception d'un port, relier sur le connecteur le signal **BB'** au signal **+POL**, et relier le signal **AA'** au signal **-POL.**

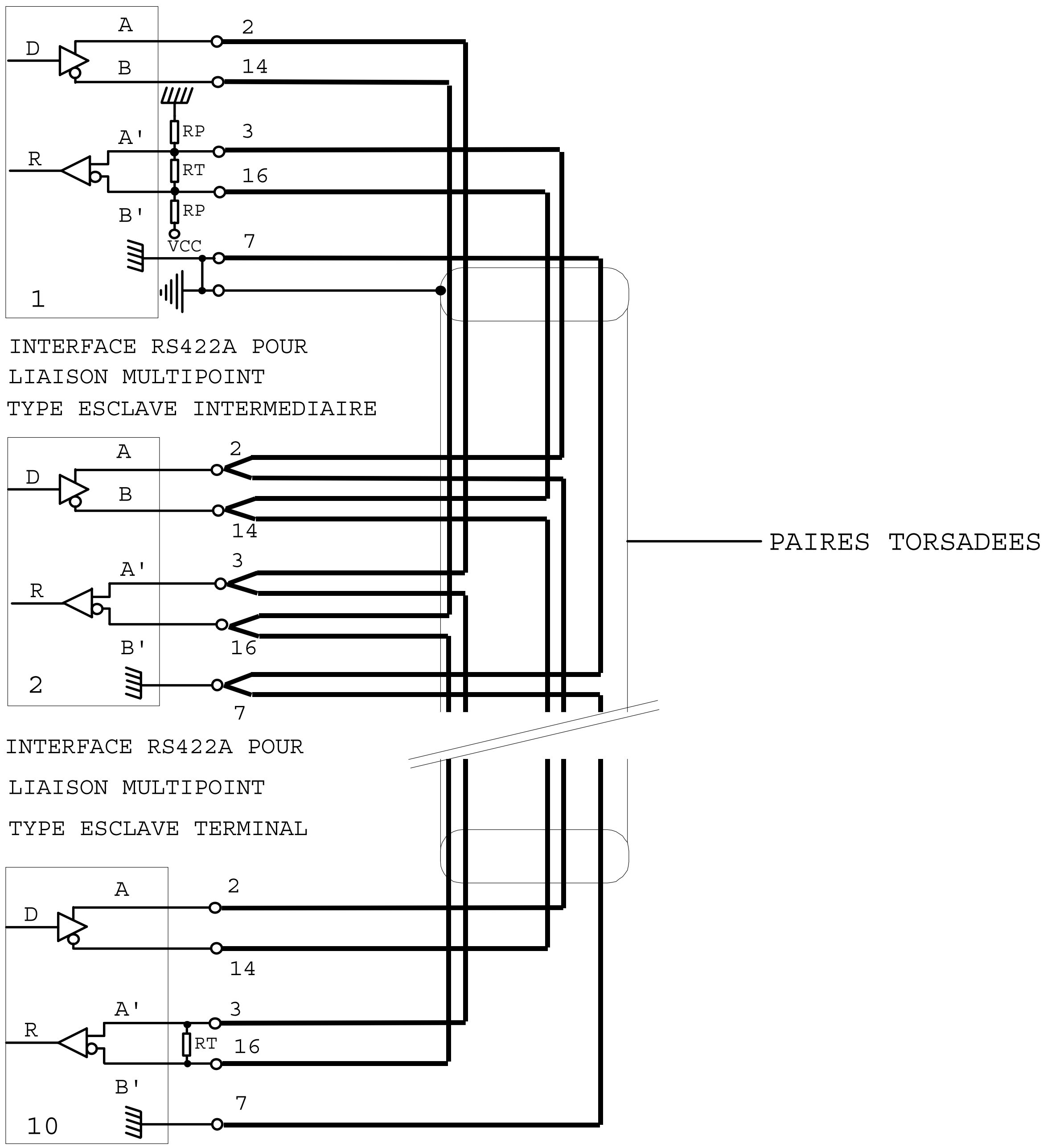
4.5.1 Exemple de câblage en RS422

CABLAGE RS422A FULL-DUPLEX POINT A POINT

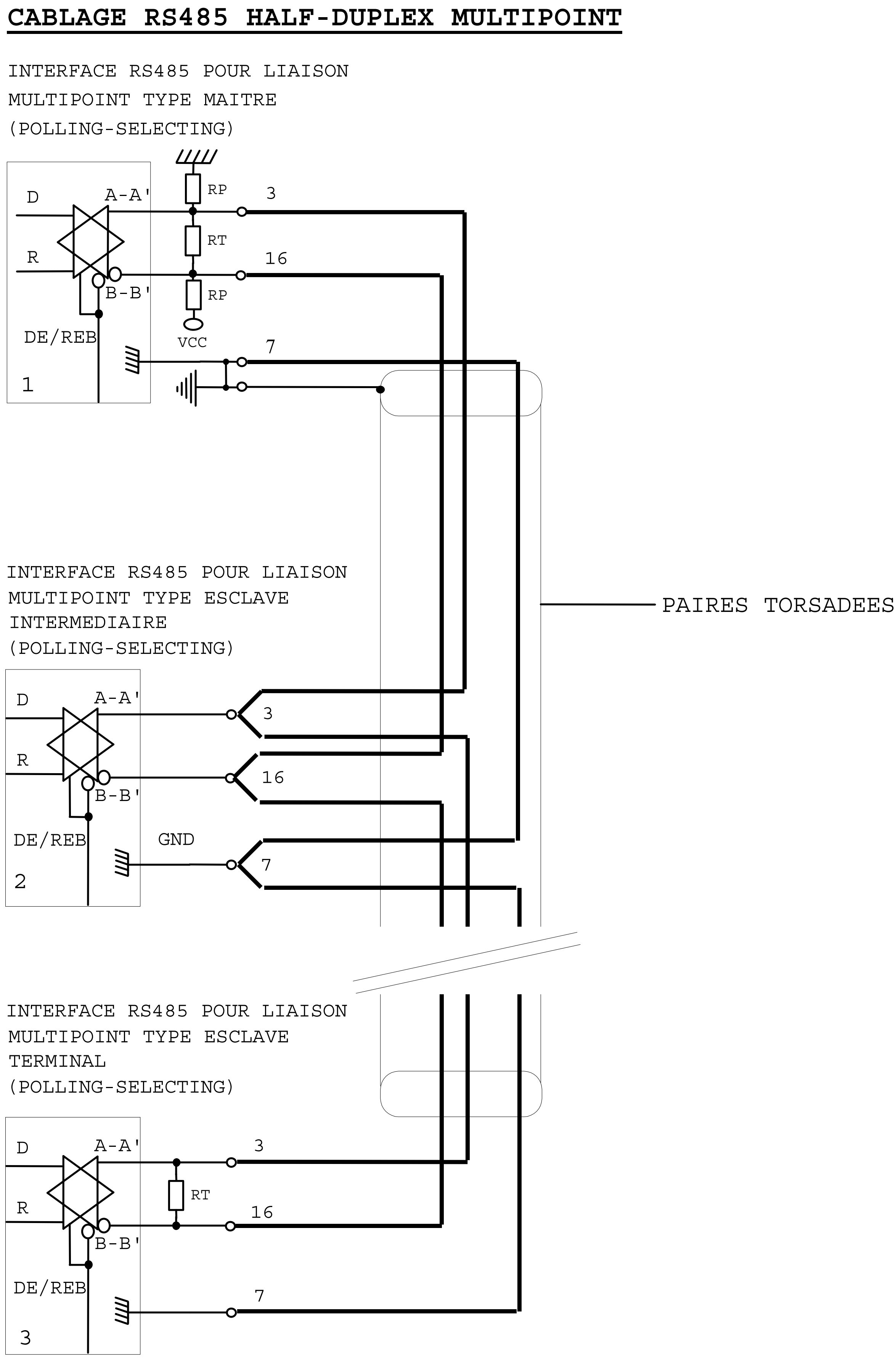


CABLAGE RS422A FULL-DUPLEX MULTIPOINT

INTERFACE RS422A POUR
LIAISON MULTIPOINT TYPE MAITRE
(POLLING SELECTING)



4.5.2 Exemple de câblage en RS485



5. INSTALLATION DU PILOTE ACKSYS POUR WINDOWS 98 & MILLENNIUM

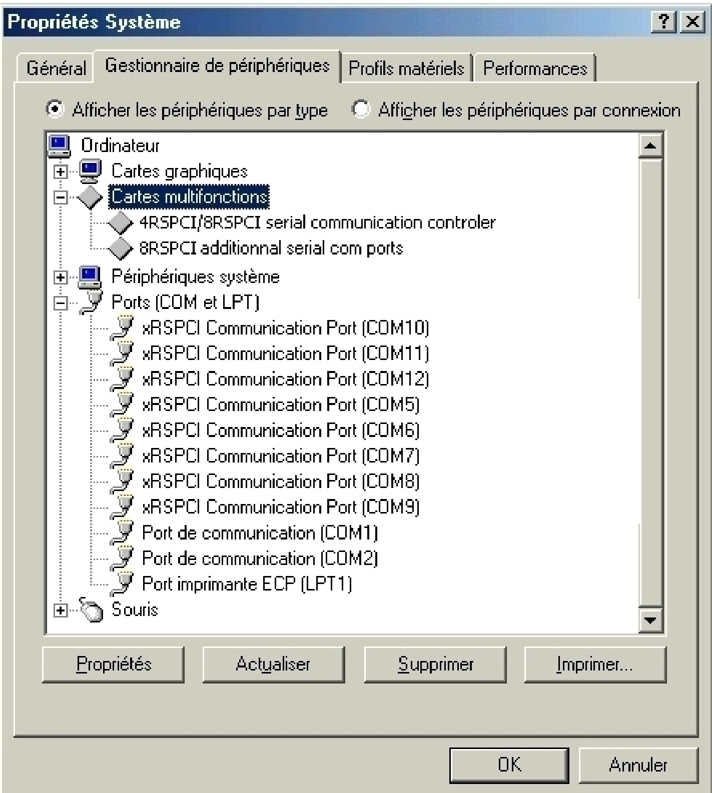
Installez la carte XRSPCI dans le PC (voir section précédente) puis démarrez Windows.
Comme il s'agit d'une carte PCI, une fois que la carte a été mise ou retirée, la configuration sera automatiquement mise à jour.

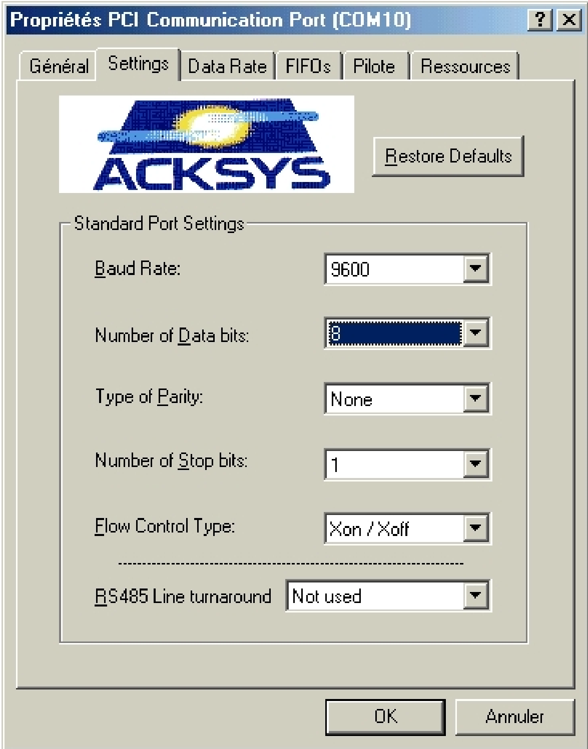
Sous Windows 98, 98 SE et Millennium, l'assistant *ajout de nouveau matériel* est automatiquement lancé lors de la première installation, dès que la carte a été détectée. Suivez les indications de l'assistant pour installer la carte.

L'installation du pilote est maintenant achevée avec succès. Si vous voulez ajouter une carte alors que le pilote a déjà été installé, Windows 98 passera l'assistant d'ajout de matériel et installera automatiquement les ports de communication PCI.

5.1 Propriétés des ports de communication

Les nouveaux ports de communication sont fonctionnels dès la fin de l'installation, et sont alors visibles dans le gestionnaire de périphérique (panneau de configuration/système). Pour éditer les propriétés d'un port, double-cliquez la ligne correspondante.





l'onglet "**Settings**" permet de définir les paramètres de communication initiaux, comme dans le cas d'un port de COM standard.

Retournement automatique de ligne en RS485 (RS485 line turnaround)

Il s'agit du pilotage du sens de communication en RS485. Si le port est configuré en RS422 ou en RS232 (liaisons point à point), l'émission et la réception simultanées ('full duplex') sont autorisées, il n'est donc pas nécessaire de piloter le sens de communication. L'option sélectionnée doit être '**Not used**' : l'émission est alors validée en permanence.

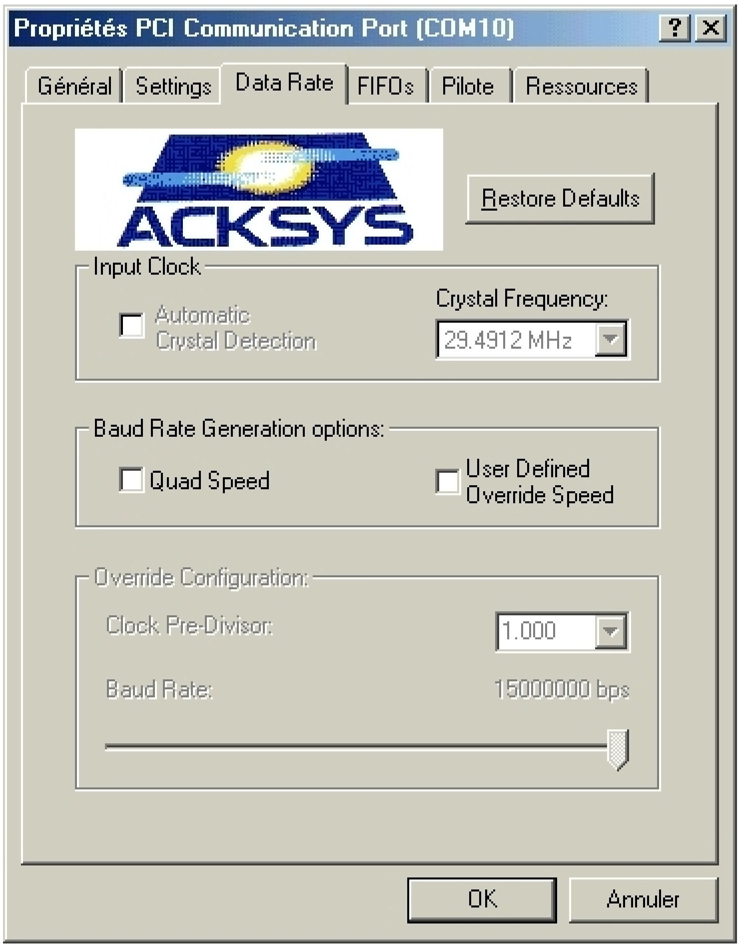
En mode RS485, l'émission et la réception sont alternées ('half duplex'), il est donc nécessaire de piloter un signal indiquant si la ligne doit être en émission ou en réception. Au repos (pas de transactions), la ligne est en réception; dès qu'un caractère ou qu'un groupe de caractères doit être émis, la ligne doit être commutée en mode émission : la carte xRSPCI utilise le signal DTR pour cette fonction.

Lorsque l'option '**Driven by application**' est sélectionnée, le pilotage du signal DTR est à la charge de l'application : le signal doit être désactivé (1 logique) avant l'émission et désactivé (0 logique) lorsque le dernier caractère a été émis. Cette méthode ne permet pas un contrôle précis du temps de commutation après émission complète de la trame.

Pour un contrôle plus efficace, ou si l'application ne peut pas gérer le signal DTR, sélectionnez l'option '**Automatic**' : le signal DTR sera alors piloté automatiquement par l'UART à chaque envoi de caractère, garantissant ainsi un temps optimal de commutation du sens de communication.



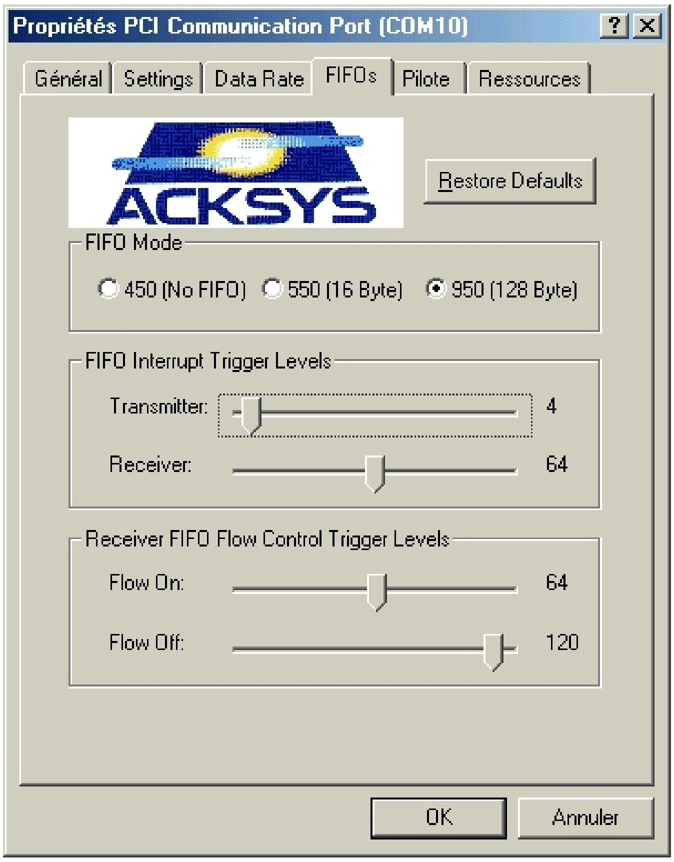
***Remarque :** Le pilotage du sens de communication peut être nécessaire en mode **RS422**, lorsque plusieurs équipements sont connectés en bus (**RS422 maître/esclave ou RS485 4 fils**). Si le port de la carte xRSPCI est un périphérique 'esclave', il partage sa ligne d'émission avec les autres périphériques 'esclaves' du réseau et doit par conséquent laisser celle ci en état 'haute impédance' lorsqu'il n'émet pas de caractères.*



L'onglet '**Data Rate**' permet de configurer des modes de fonctionnement spécifiques de la carte et de visualiser la fréquence de l'oscillateur, qui doit être, pour les cartes xRSPCI standards, de 29,4912 MHz

- Option '**Quad Speed**' : cette fonctionnalité ne doit pas être utilisée pour les applications conventionnelles, laissez la case non cochée.

- Option '**User Defined Override Speed**' : cette option permet de forcer la vitesse de communication fixée dans le cadre '**Override Configuration'**. Dans ce cas, le pilote de la carte xRSPCI ignore le paramètre de vitesse passé par l'API Windows. Le '**Clock Pre-Divisor**' permet d'appliquer un facteur de division de l'horloge pour obtenir une meilleure précision lorsqu'une vitesse non standard élevée est requise.



L'onglet **FIFOs** permet de fixer les seuils de déclenchement des interruptions d'émission et de réception en fonction du nombre de caractères présents dans les tampons respectifs, ainsi que les seuils pour le contrôle de flux. Les valeurs par défaut sont satisfaisantes pour la plupart des applications classiques.

Réglages des seuils d'interruption :

La valeur définie à l'aide du curseur 'Transmitter' indique le seuil à partir duquel une interruption d'émission sera générée. Par exemple, la valeur par défaut, 4, indique qu'une interruption sera générée dès que le nombre de caractères restants dans la FIFO de transmission passera de 5 à 4. Cette valeur devrait rester petite, mais il peut être souhaitable de l'augmenter aux vitesses élevées ou avec des unités centrales peu puissantes ou chargées.

La valeur définie à l'aide du curseur 'Receiver' indique le seuil à partir duquel une interruption de réception sera générée. Dans le cas de la valeur par défaut, l'interruption se produit lorsque le nombre de caractères présent dans la FIFO de réception passe de 63 à 64. Si le nombre de caractères reçus est inférieur au seuil et n'évolue pas pendant un temps correspondant au temps d'émission de 4 caractères, une interruption 'time out' est générée de façon à avertir le pilote de périphérique.

Dans le cas d'une application effectuant des transferts de blocs de données importants, on choisira de préférence des seuils élevés de façon à réduire le nombre d'interruptions et par conséquent le taux d'occupation CPU. Il n'est pas conseillé cependant d'utiliser les valeurs maximales, notamment lorsque la vitesse de communication est élevée, afin d'éviter les écrasements en réception et les interruptions de l'émission.

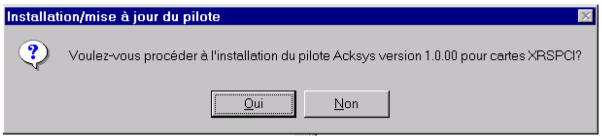
6. INSTALLATION DU PILOTE ACKSYS POUR WINDOWS NT 4.0

Installez la carte xRSPCI (voir section 3) et démarrez Windows NT.

Pour installer le pilote :

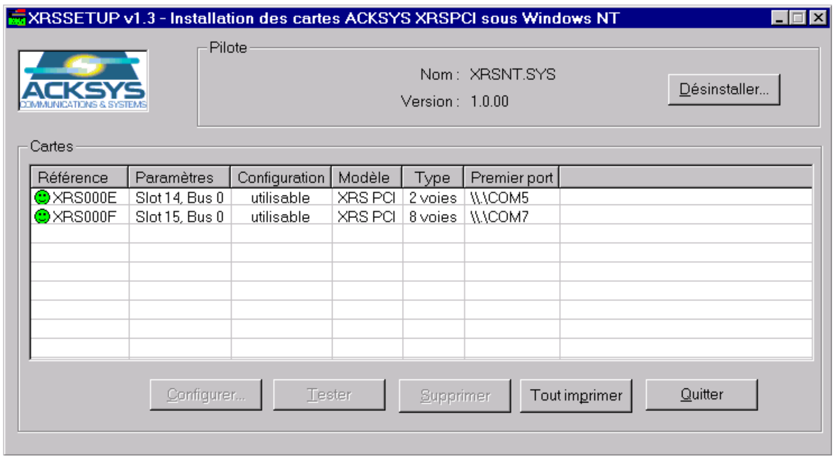
Démarrez votre PC et identifiez vous comme *Administrateur*

Insérez la disquette[1] nommée xRSPCI drivers

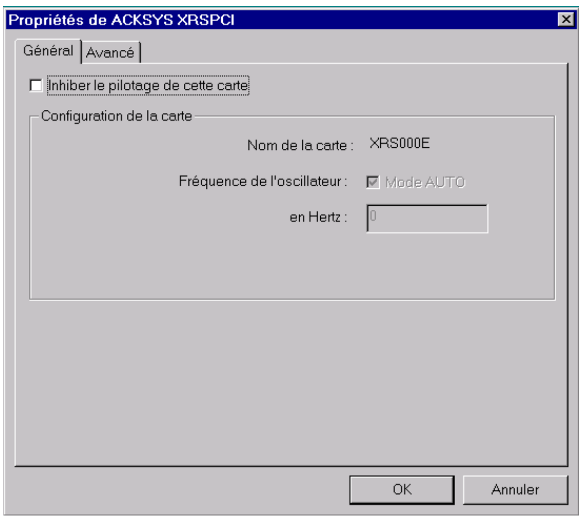


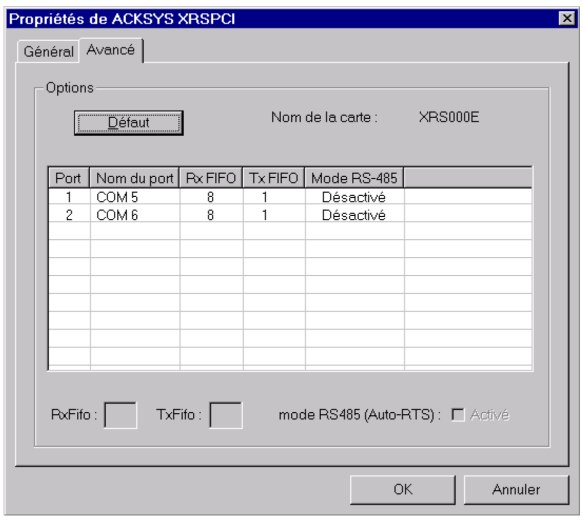
Lancez le programme XRSSETUP à partir de A:\WINNT

Cliquez sur le bouton ***Oui***



Dans cette fenêtre, vous verrez apparaître la liste des cartes xRSPCI détectées par le pilote. Vous pouvez alors cliquer sur le bouton **Quitter** pour terminer l'installation ou sélectionner une carte et cliquer sur le bouton "**Configurer …**" pour ouvrir la fenêtre *Propriétés de ACKSYS xRSPCI.*



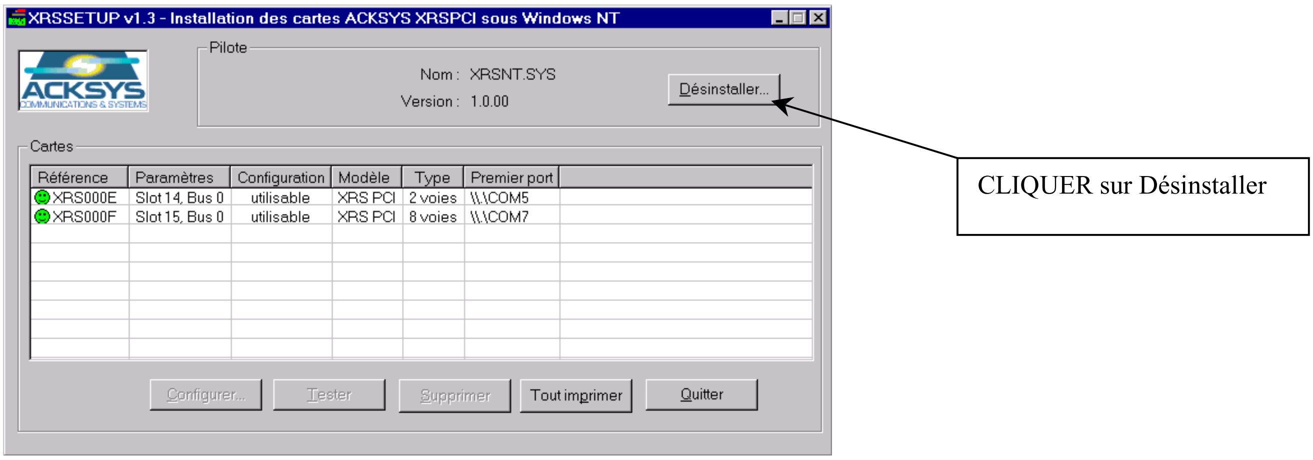


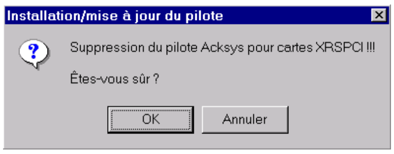
Dans la fenêtre *Propriétés de ACKSYS XRSPCI* :

Vous pouvez désactiver une carte en cliquant sur la case *« Inhiber le pilotage de cette carte ».*

Vous pouvez spécifier, pour chaque port, le niveau de déclenchement du RxFIFO, la taille du TxFIFO et le mode RS485.

Pour désinstaller le pilote et la carte xRSPCI , lancez XRSSETUP.EXE et cliquez sur le bouton « Désinstaller ».







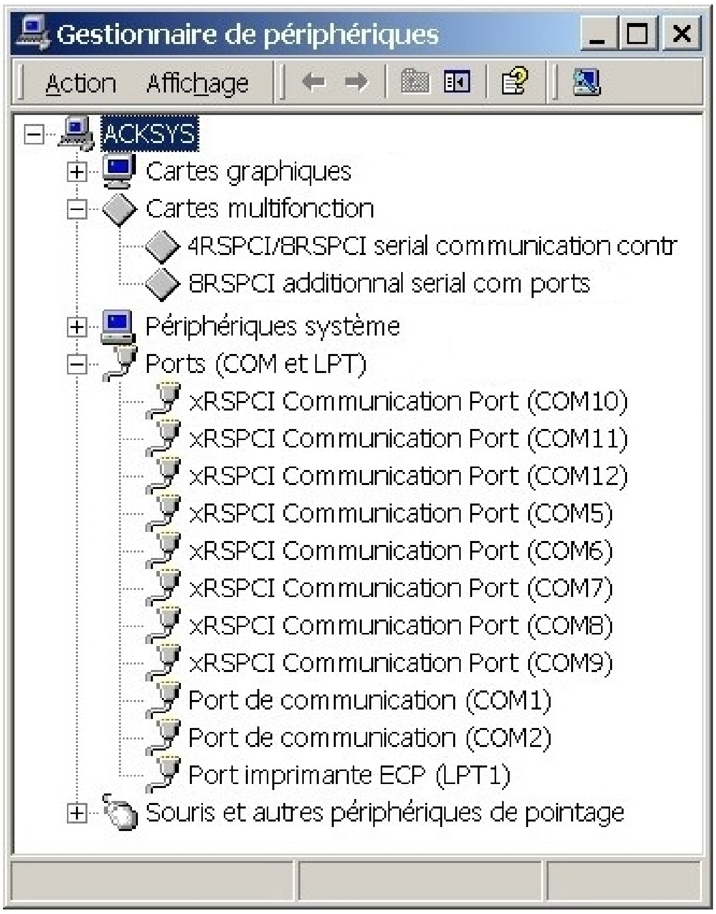
[1] Les drivers pourront éventuellement être fournis sur CD. Dans ce cas, il conviendra de sélectionner le répertoire contenant les drivers xRSPCI.

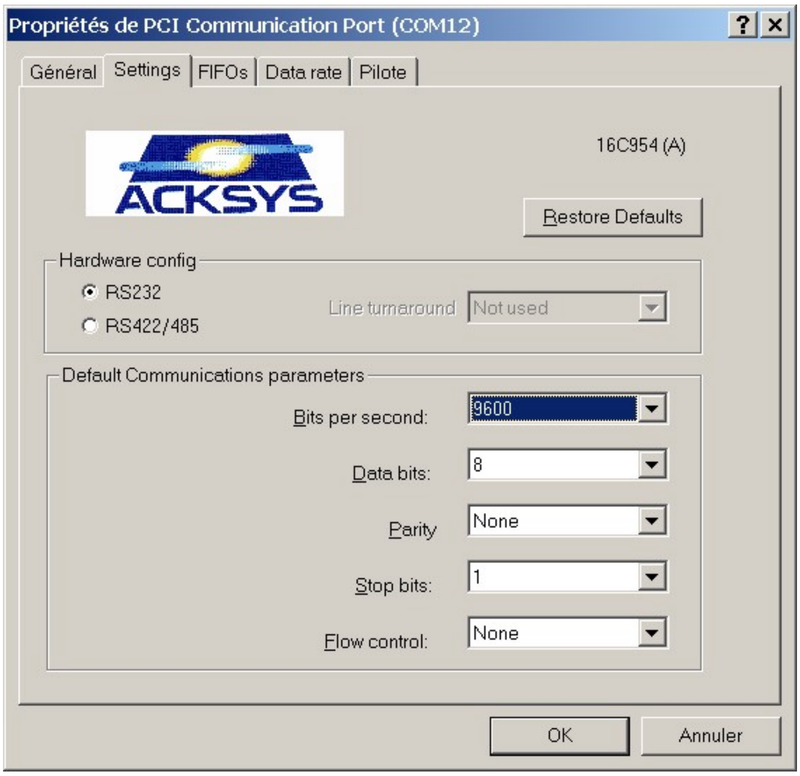
7. INSTALLATION DU PILOTE ACKSYS POUR WINDOWS 2000/XP

Comme Windows 9x ou ME, Windows 2000 et Windows XP détectent automatiquement la carte XRSPCI. Un assistant d'installation de matériel est automatiquement lancé au démarrage du système, dès que la carte a été détectée. Suivez les indications de l'assistant pour installer la carte.
NOTE POUR WINDOWS XP : L'assistant va vous signaler que le pilote de périphérique n'est pas certifié. Ignorez ce message en cliquant sur le bouton « continuer » afin de poursuivre l'installation de la carte.

7.1 Propriétés des ports de communication

Les nouveaux ports de communication sont visibles dans le gestionnaire de périphériques (panneau de configuration / matériel). Double-cliquez sur un des ports pour éditer ses propriétés.





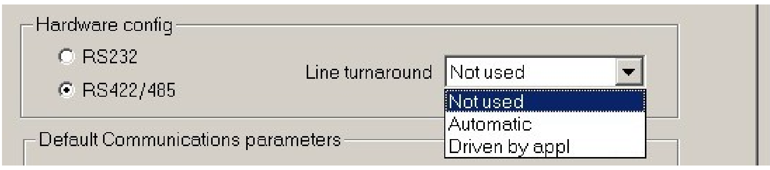
L'onglet "**Settings**" permet de définir les paramètres de communication par défaut, comme dans le cas d'un port de COM standard. Il peut être utilisé également pour définir le type d'interface et valider le retournement automatique de ligne en RS485.

Type d'interface

RS232 : Dans ce mode le contrôle de flux hardware et le positionnement des signaux de contrôle sont autorisés par le pilote de périphérique. Le retournement automatique de ligne est interdit.

RS422/485 : Dans ce mode, le contrôle de flux hardware est interdit. Il est possible de programmer le retournement automatique de ligne.

Retournement de ligne ('line turnaround')



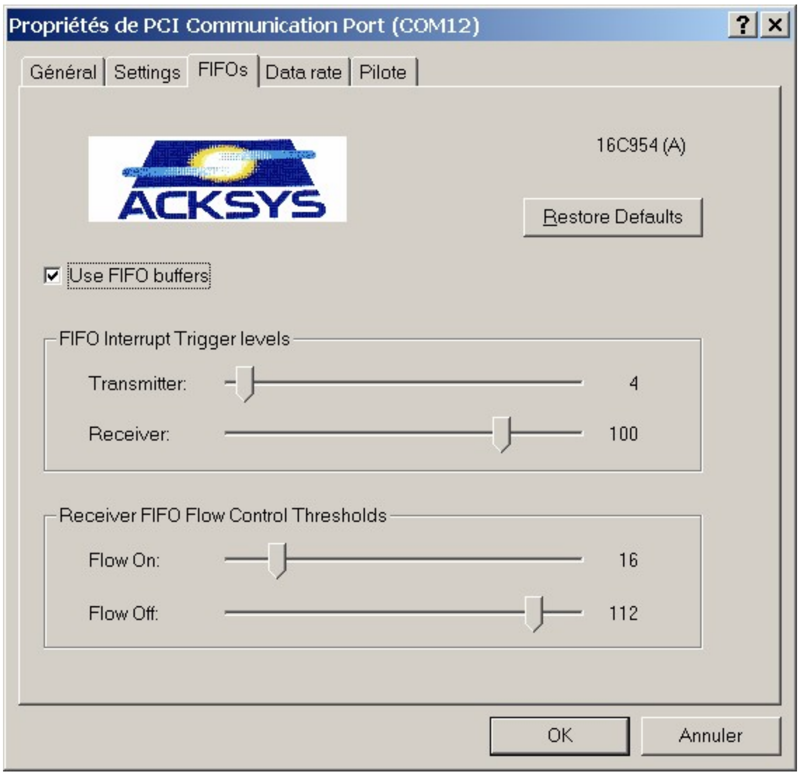
Il s'agit du pilotage du sens de communication en RS485. Si le port est configuré en RS422 (liaison point à point), l'émission et la réception simultanées ('full duplex') sont autorisées, il n'est donc pas nécessaire de piloter le sens de communication : l'option sélectionnée doit être '**Not used**' : l'émission est alors validée en permanence.

En mode RS485, l'émission et la réception sont alternées ('half duplex'). Il est donc nécessaire de piloter un signal indiquant si la ligne doit être en émission ou en réception. Au repos (pas de transactions), la ligne est en réception; dès qu'un caractère ou qu'un groupe de caractères doit être émis, la ligne doit être commutée en mode émission : la carte xRSPCI utilise le signal DTR pour cette fonction.

Lorsque l'option '**Driven by application**' est sélectionnée, le pilotage du signal DTR est à la charge de l'application : le signal doit être désactivé (1 logique) avant l'émission et activé (0 logique) lorsque le dernier caractère a été émis. Cette méthode ne permet pas un contrôle précis du temps de commutation après émission complète de la trame.

Pour un contrôle plus efficace, ou si l'application ne peut pas gérer le signal DTR, sélectionnez l'option '**Automatic**' : le signal DTR sera alors piloté automatiquement par l'UART à chaque envoi de caractère, garantissant ainsi un temps optimal de commutation du sens de communication.

Remarque : *Le pilotage du sens de communication peut être nécessaire en mode* ***RS422****, lorsque plusieurs équipements sont connectés en bus (****RS422 maître/esclave ou RS485 4 fils****). Si le port de la carte xRSPCI est un périphérique 'esclave', il partage sa ligne d'émission avec les autres périphériques 'esclaves' du réseau et doit par conséquent laisser celle ci en état 'haute impédance' lorsqu'il n'émet pas de caractères.*



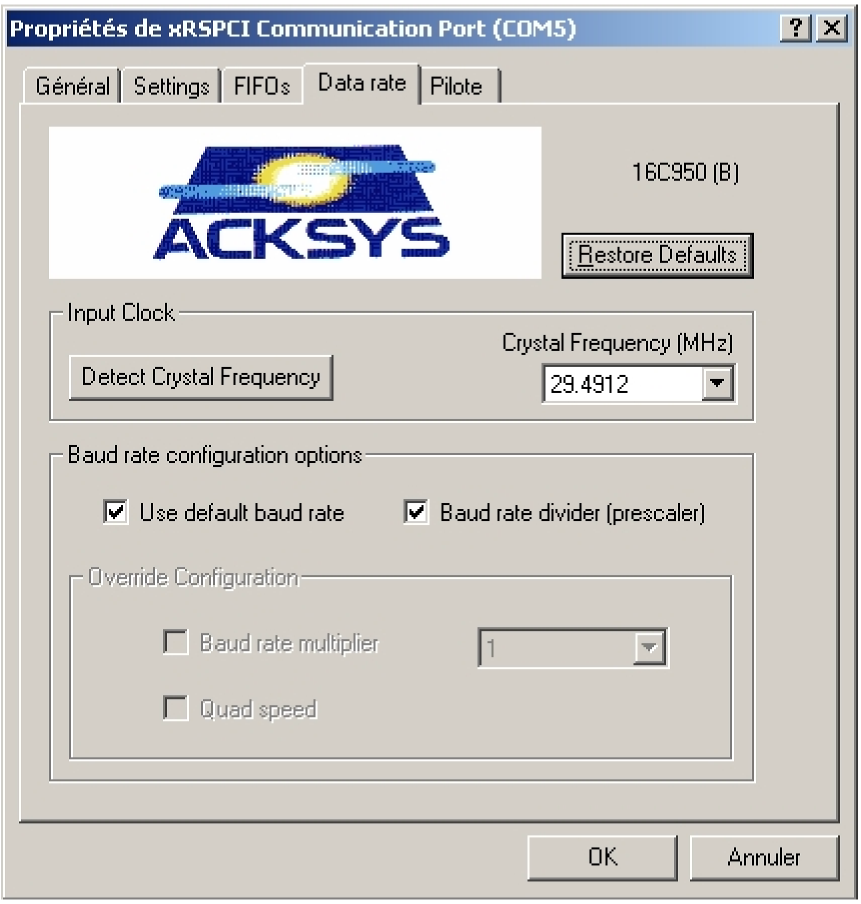
L'onglet **FIFOs** permet de fixer les seuils de déclenchement des interruptions d'émission et de réception en fonction du nombre de caractères présents dans les tampons respectifs, ainsi que les seuils pour le contrôle de flux. Les valeurs par défaut sont satisfaisantes pour la plupart des applications classiques.

Réglages des seuils d'interruption :

La valeur définie à l'aide du curseur 'Transmitter' indique le seuil à partir duquel une interruption d'émission sera générée. Par exemple, la valeur par défaut, 4, indique qu'une interruption sera générée dès que le nombre de caractères restants dans la FIFO de transmission passera de 5 à 4. Cette valeur devrait rester petite, mais il peut être souhaitable de l'augmenter aux vitesses élevées ou avec des unités centrales peu puissantes ou chargées.

La valeur définie à l'aide du curseur 'Receiver' indique le seuil à partir duquel une interruption de réception sera générée. Dans le cas de la valeur par défaut, l'interruption se produit lorsque le nombre de caractères présent dans la FIFO de réception passe de 63 à 64. Si le nombre de caractères reçus est inférieur au seuil et n'évolue pas pendant un temps correspondant au temps d'émission de 4 caractères, une interruption 'time out' est générée de façon à avertir le pilote de périphérique.

Dans le cas d'une application effectuant des transferts de blocs de données importants, on choisira de préférence des seuils élevés de façon à réduire le nombre d'interruptions et par conséquent le taux d'occupation CPU. Il n'est pas conseillé cependant d'utiliser les valeurs maximales, notamment lorsque la vitesse de communication est élevée, afin d'éviter les écrasements en réception et les interruptions de l'émission.



L'onglet '**Data Rate**' permet de sélectionner la fréquence de l'oscillateur pour certains modèles spécifiques de cartes xRSPCI. La valeur standard par défaut est 29,4912 MHz.

- Option '**Use default baud rate**' : Doit rester sélectionnée. Elle indique que la vitesse de communication est fixée de façon classique par l'API Win32.

- Option '**Baud rate divider (prescaler)**' : cette option est activée par défaut et autorise le calcul de la vitesse de communication avec utilisation d'un prescalaire décimal permettant d'obtenir une meilleure précision pour les vitesses non standards. Dans ce cas, le pilote de périphérique de la carte xRSPCI calcule les paramètres donnant la précision optimale par rapport à la vitesse demandée. Lorsque cette option n'est pas sélectionnée, les arrondis des vitesses non standard sont compatibles avec des cartes plus anciennes.

- Option '**Baud rate multiplier**' : cette fonctionnalité ne peut pas être sélectionnée lorsque la case 'Use default baud rate' est cochée. Elle permet d'appliquer le facteur de multiplication choisi dans la boite de sélection à la vitesse demandée par l'application. Par exemple avec un facteur de 16, lorsque l'application demande une vitesse de 115200 bauds, la vitesse effective sera de 1,8432MHz, autorisant ainsi une application qui serait limitée à 115200 bauds à travailler à des vitesses plus élevées.

- Option '**Quad Speed**' : cette fonctionnalité ne peut pas être sélectionnée lorsque la case 'Use default baud rate' est cochée. Elle ne doit pas être utilisée pour les applications conventionnelles, laissez la case non cochée.

8. INSTALLATION DU PILOTE ACKSYS V2.1 POUR LINUX (NOYAU 2.2.17-21)

La procédure d'installation a été testée initialement sur **Linux Mandrake** version **7.2** et une version du noyau **2.2.17-21**. Pour tout problème de compatibilité avec d'autres versions de Linux, merci de contacter ACKSYS. Tous les programmes décrits ci-dessous se trouvent sous /linux/V2.1 sur le support de distribution.

Installation du module

Ce type d'installation n'oblige pas une recompilation de Linux. De plus, elle permet de charger et décharger le driver de façon dynamique.

- Copier le fichier **`srllinux.o`** dans le répertoire

  `/lib/modules/$(shell uname -r)/misc/`

- Il faut créer les nœuds dans le système. Pour cela, éditez le script **msmknod** fournit par ACKSYS, ajustez la constante **MAXPORT** suivant votre carte (si vous avez une carte deux voies mettre 2, si vous avez une carte 4 voies mettre 4 ...) puis exécutez ce script. Il va créer les fichiers de périphérique de la façon suivante :

Carte n°	*Numéro de périphérique*
1	**`ttyM0-ttyM[MAXPORT-1]`**
2	**`ttyM[MAXPORT]-ttyM[2*MAXPORT-1]`**

- Pour installer le module il suffit de faire **`insmod srllinux`**
- Pour arrêter le module il suffit de faire **`rmmod srllinux`**
- Pour vérifier que le module est bien installé, utilisez la commande **`lsmod`**.

REMARQUE :

Au prochain redémarrage de Linux, le module ne se lancera pas automatiquement, pour cela ajoutez le fichier **`rc.ack`** dans le fichier **`rc.serial`**.

Si le fichier **`rc.serial`** n'existe pas créez le, ajoutez la ligne :

```
/etc/rc.d/rc.ack
```

Puis éditez le fichier **`rc.local`**, ajoutez la ligne suivante :

```
if [ -f /etc/rc.d/rc.serial ]; then
     Sh /etc/rc.d/rc.serial
fi
```

NOTE

Si l'oscillateur de la carte n'est pas à 29,4912 MHz, lorsque vous installez le module vous devez ajouter l'option :

`input_clock=[Freq. en Hz]`.

Par exemple, pour un oscillateur à 16 MHz, la commande est la suivante :

```
insmod srllinux input_clock=16000000
```

9. INSTALLATION DU PILOTE ACKSYS V3.X POUR LINUX (NOYAU 2.4.X)

La procédure d'installation a été testée initialement sur **Linux Mandrake** version **9.0** avec une version du noyau 2.4.19, ainsi que sur **Linux redhat** 7.2 avec un noyau 2.4.7-10. Pour tout problème de compatibilité avec d'autres versions de Linux, merci de contacter ACKSYS.
Ce driver est basé sur le Driver Linux Serial 5.05.
Tous les programmes décrits ci-dessous se trouvent sous /linux/V3.4 sur le support de distribution.

9.1 Installation du module

Ce type d'installation n'oblige pas une recompilation de Linux. De plus, elle permet de charger et décharger le driver de façon dynamique.
Copier le fichier **`srlxrspci.o`** dans le répertoire

    /lib/modules/$(shell uname -r)/misc/

Les fichiers de périphérique peuvent être créés avec les noms standard (ttyS) ou avec un nom choisi au moment de l'installation (ex : ttyA).

9.2 Démarrage du driver

Pour démarrer le driver, il faut taper la commande :

    insmod srlxrspci.o

Cela vient créer les tty dans le système. Par défaut, le driver va créer les terminaux portant le nom ttyA04 jusqu'à ttyAn (n dépendant de la carte).
Si vous souhaitez changer le nom du terminal, il suffit de passer en argument

    tty_name_p=<nom du terminal>

Par exemple si vous voulez créer un terminal avec le nom ttyS, tapez :

    insmod srlxrspci.o tty_name_p=ttyS

Le driver va créer les terminaux ttyS04 jusqu'à ttySn (n dépendant de votre carte).

Remarque :

Pour vérifier le nom du terminal qui est créé, regarder le fichier `/var/log/messages`.

9.3 Création des nœuds dans le système

Une fois le driver correctement démarré, Il faut créer les nœuds dans le système. Pour cela taper la commande :

    mknod <nom terminal> c <major> <minor>

Ce qui donne, pour un démarrage avec les options par défaut :

    mknod ttyA04 c 40 68

Remarque :

Pour vérifier le numéro major et minor du terminal crée par le driver, regarder le fichier `/var/log/messages`.

9.4 arrêt du driver

- Pour arrêter le module il suffit de faire **`rmmod srlxrspci`**
- Pour vérifier que le module est bien installé, utilisez la commande **`lsmod`**.

9.5 Remarques sur le driver

✓ Au prochain redémarrage de Linux, le module ne se lancera pas automatiquement, Pour cela ajoutez le fichier `rc.ack` dans le fichier `rc.serial`.

Si le fichier `rc.serial` n'existe pas créez le, ajouter la ligne :

```
/bin/sh /etc/rc.d/rc.ack
```

Puis éditez le fichier `rc.local`, ajoutez la ligne suivante :

```
if [ -f /etc/rc.d/rc.serial ]; then
     /bin/sh /etc/rc.d/rc.serial
fi
```

La procédure décrite ci-dessus peut être différente sur une autre version de Linux.
Vérifiez que dans le fichier rc.ack le driver est bien lancé avec les bonnes options.

✓ Si l'oscillateur de la carte n'est pas à 29,4912 MHz, lorsque vous installez le module vous devez ajouter l'option :

input_clock=[Freq. en Hz].

Par exemple, pour un oscillateur à 16 MHz, la commande est la suivante :

```
insmod srlxrspci input_clock=16000000
```

✓ Si vous voulez utiliser une vitesse non standard, lorsque vous installez le module vous devez ajouter l'option :

speed_custom=[vitesse en bauds]

Par exemple, si vous souhaitez une vitesse de 76800 bauds, la commande est la suivante :

```
insmod srlxrspci.o speed_custom=76800
```

9.6 Utilisation du paramètre speed_custom

Pour configurer la vitesse d'un port série, les fonctions à utiliser sont :

```
int cfsetospeed(struct termios *termios_p, speed_t speed); // vitesse de sortie
int cfsetispeed(struct termios *termios_p, speed_t speed); // vitesse d'entrée
```

Lorsque vous voulez prendre en compte le paramètre `speed_custom`, vous devez utiliser la constante **EXTA**, comme le montre l'exemple ci-dessous :

```
cfsetospeed(&ma_struct_termios, EXTA);
```

Cette constante **EXTA** est définie par défaut dans le fichier `/usr/include/bits/termios.h` de la façon suivante :

```
#define EXTA B19200
```

Si vous souhaitez utiliser la vitesse 19200 bauds (B19200), vous devez modifier la constante **EXTA** dans `/usr/include/bits/termios.h` et il faudra alors recompiler le driver Linux.

9.7 Références

Fichier d'aide linux relatif aux ports série :
http://en.tldp.org/HOWTO/Serial-HOWTO.html

Fichier d'aide linux relatif à la programmation des ports série :
http://en.tldp.org/HOWTO/Serial-Programming-HOWTO/

Liste des sites contenant les Linux Howtos :
http://metalab.unc.edu/LDP/mirrors.html

Fichier source du driver série linux 5.05 :
http://sourceforge.net/projects/serial
http://sourceforge.net/project/showfiles.php?group_id=310

10. UTILISATION DU MODE RS485 SOUS LINUX

Lorsque l'on utilise une ligne en mode RS485, il est possible de gérer le retournement de façon automatique. Pour cela il faut utiliser les *iocontroles* suivants :

- **ACKSYS_ENABLE_485_MODE** : Valide la fonction de prise de ligne automatique.
- **ACKSYS_DISABLE_485_MODE** : Dévalide la fonction de prise de ligne automatique.

La prise de ligne automatique sera gérée par l'UART. Une fois le mode 485 activé, il le restera jusqu'à l'envois de la commande de désactivation.
Ces IOCTLs ne prennent pas de paramètre. La fonction de l'API linux permettant d'envoyer l'iocontrole au driver est ioctl(…). Cette fonction est à envoyer une fois le port de communication ouvert. Si vous fonctionnez en mode RS485 il ne faut pas envoyer l'iocontrol ACKSYS_DISABLE_485_MODE, sous risque de perturber le fonctionnement du bus. Cette iocontrol ne sert que dans le cas ou vous voulez passer du mode RS485 au mode RS422/RS232.
Les constantes associées à ces 2 IOCTLs sont définies dans le fichier iocontrol.h.

ATTENTION : La valeur des deux constantes n'est pas identique sur les deux noyaux linux. Il vous faudra obligatoirement recompiler votre application avec le bon fichier include.

11. ANNEXE A - LE CONNECTEUR 62 POINTS (CARTES 4 ET 8 PORTS)

11.1 Cartes en version RS232

Pin	RS232	Pin	RS232	Pin	RS232
1	DCD1	**22**	RXD1	**43**	GND
2	TXD1	**23**	DTR1	**44**	CTS1
3	DSR1	**24**	RTS1	**45**	TXD2
4	DCD2	**25**	RXD2	**46**	GND
5	DTR2	**26**	DSR2	**47**	DCD3
6	RTS2	**27**	CTS2	**48**	DTR3
7	RXD3	**28**	TXD3	**49**	GND
8	DSR3	**29**	RTS3	**50**	RXD4
9	CTS3	**30**	DCD4	**51**	GND
10	TXD4	**31**	DTR4	**52**	CTS4
11	DSR4	**32**	RTS4	**53**	TXD5
12	DCD5	**33**	RXD5	**54**	GND
13	DTR5	**34**	DSR5	**55**	DCD6
14	RTS5	**35**	CTS5	**56**	DTR6
15	RXD6	**36**	TXD6	**57**	GND
16	DSR6	**37**	RTS6	**58**	RXD7
17	CTS6	**38**	DCD7	**59**	DSR7
18	TXD7	**39**	DTR7	**60**	DCD8
19	RTS7	**40**	CTS7	**61**	DTR8
20	RXD8	**41**	TXD8	**62**	CTS8
21	DSR8	**42**	RTS8		

Tableau 6.1. Attribution des signaux sur le connecteur 62 points en mode RS232

11.2 Cartes en version RS422

Pin	RS422	Pin	RS422	Pin	RS422
1	-pol1	**22**	A'1	**43**	GND
2	A1	**23**	+pol1	**44**	B'1
3	n.c.	**24**	B1	**45**	A2
4	-pol2	**25**	A'2	**46**	GND
5	+pol2	**26**	n.c.	**47**	-pol3
6	B2	**27**	B'2	**48**	+pol3
7	A'3	**28**	A3	**49**	GND
8	n.c.	**29**	B3	**50**	A'4
9	B'3	**30**	-pol4	**51**	GND
10	A4	**31**	+pol4	**52**	B'4
11	n.c.	**32**	B4	**53**	A5
12	-pol5	**33**	A'5	**54**	GND
13	+pol5	**34**	n.c.	**55**	-pol6
14	B5	**35**	B'5	**56**	+pol6
15	A'6	**36**	A6	**57**	GND
16	n.c.	**37**	B6	**58**	A'7
17	B'6	**38**	-pol7	**59**	n.c.
18	A7	**39**	+pol7	**60**	-pol8
19	B7	**40**	B'7	**61**	+pol8
20	A'8	**41**	A8	**62**	B'8
21	n.c.	**42**	B8		

Tableau 6.2. Attribution des signaux sur le connecteur 62 points en mode RS422

Polarisation et résistances de terminaison :

Toutes les voies disposent de résistances de terminaison et de polarisation sur leur paire de réception (A'B'). Il est possible de supprimer ces résistances en ôtant les réseaux RS16 (pour les voies 1 à 4) et RS40 (pour les voies 5 à 8). Dans ce cas, il reste possible de polariser individuellement les voies en utilisant les sorties de polarisation disponibles sur le connecteur 62 points (J5) : relier **+pol**$_n$ à **B'**$_n$ et relier **-pol**$_n$ à **A'**$_n$.

Par exemple, pour polariser la voie **3**, relier **+pol**$_3$ **(48)** à **B'**$_3$ **(9)** puis **–pol**$_3$ **(47)** à **A'**$_3$ **(7)**

Si des résistances de terminaison sont montées individuellement, elles devront être câblées à l'extérieur.

11.3 Cartes en version RS485

Pin	RS485
1	-pol1
2	réservé
3	n.c.
4	-pol2
5	+pol2
6	réservé
7	AA’3
8	n.c.
9	BB’3
10	réservé
11	n.c.
12	-pol5
13	+pol5
14	réservé
15	AA’6
16	n.c.
17	BB’6
18	réservé
19	réservé
20	AA’8
21	n.c.

Pin	RS485
22	AA’1
23	+pol1
24	réservé
25	AA’2
26	n.c.
27	BB’2
28	réservé
29	réservé
30	-pol4
31	+pol4
32	réservé
33	AA’5
34	n.c.
35	BB’5
36	réservé
37	réservé
38	-pol7
39	+pol7
40	BB’7
41	réservé
42	réservé

Pin	RS485
43	GND
44	BB’1
45	réservé
46	GND
47	-pol3
48	+pol3
49	GND
50	AA’4
51	GND
52	BB’4
53	réservé
54	GND
55	-pol6
56	+pol6
57	GND
58	AA’7
59	n.c.
60	-pol8
61	+pol8
62	BB’8

Tableau 2. Attribution des signaux sur le connecteur 62 points en mode RS485

Polarisation & résistances de terminaison :

Toutes les voies disposent de résistances de terminaison et de polarisation. Il est possible de supprimer ces résistances en ôtant les réseaux RS16 (pour les voies 1 à 4) et RS40 (pour les voies 5 à 8). Dans ce cas, il reste possible de polariser individuellement les voies en utilisant les sorties de polarisation disponibles sur le connecteur 62 points (J5) : relier **+pol$_n$** à **BB’$_n$** et relier **-pol$_n$** à **AA’$_n$**

Par exemple, pour polariser la voie **3**, relier **+pol$_3$ (48)** à **BB’$_3$(9)** puis **–pol$_3$(47)** à **AA’$_3$(7)**

Si des résistances de terminaison sont montées individuellement, elles devront être câblées à l’extérieur.

12. ANNEXE B - PROBLEMES RENCONTRES

La carte XRSPCI n'est pas détectée par le BIOS

Vérifiez l'option PCI/PnP dans le programme setup de la carte mère, et mettez-la en AUTO.
Vérifiez que la carte est bien insérée dans le slot.
Essayez d'autres slots jusqu'à ce que vous en trouviez un bon. En cas d'échec , essayez la carte dans un autre PC pour valider son fonctionnement. Le cas échéant, consultez le constructeur du PC pour obtenir une mise à jour du BIOS.

La carte XRSPCI n'est pas détectée par Windows 95/98/2000/Millenium/Xp

Vérifiez le premier problème.
Vérifiez dans la fenêtre de Propriétés système si la carte XRSPCI n'est pas déjà reconnue en tant que carte PCI standard ou carte adaptatrice multifonction. Si c'est le cas, supprimez l'entrée correspondante et cliquez sur le bouton rafraîchir jusqu'à ce que l'assistant d'ajout de matériel soit lancé.
Réinstallez Windows.

La communication entre la carte XRSPCI et votre équipement ne fonctionne pas

Vérifiez la connexion entre votre équipement et la carte XRSPCI.
Vérifiez les paramètres de communication (vitesse, parité, nombre de bits de stops, contrôle de flux) de chaque côté.
En mode RS422/RS485, l'utilisation de l'appellation +/- pour réaliser le câblage peut ne pas fonctionner. Ceci est du à un défaut de normalisation, un constructeur appellera donc + ce qu'un autre appellera -. Il convient alors dans ce cas de relier le signal + sur le signal -. Une autre méthode consiste à identifier côté équipement les signaux A, B, A', B' ou AA' et BB' et de réaliser ensuite le câble comme indiqué dans les paragraphes 4.2 et 4.3. Notez que le signal A doit avoir une tension inférieure au signal B dans l'état MARK (état de repos).

Si vous pensez que la carte ou le logiciel présente un problème, contactez ACKSYS

- Par hotline support@acksys.fr ou par fax ++ 33 1 39 11 47 96 en décrivant votre problème dans la fiche « compte rendu d'anomalie » présente à la fin du manuel.

- Par téléphone ++ 33 1 39 11 62 81

13. XRSPCI COMPTE RENDU D'ANOMALIE

Nom	
Société	
Tel	
Fax	
Email	
XRSPCI numéro de série	
Système d'exploitation	
version du pilote ACKSYS	
Type d'ordinateur	

Description du problème

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....

....